



Robot Car

นายธัญธร พรสวัสดิ์ชัย 61010497

นายพิพิธพงศ์ จิตภักดีไทย 61010750

Function :

MOVE(int, int);

เราจะใส่พารามิเตอร์สองตัวเข้าไปคือค่าความเร็วของมอเตอร์ซ้ายและขวา โดยถ้าเราใส่ค่าติดลบ มันจะทำการเปลี่ยนมอเตอร์ให้กลายเป็นถอยหลังเองโดยอัตโนมัติ

```
void MOVE(int l, int r)
{
    if(l>0)
    {
        analogWrite(enA,l);
        digitalWrite(in2,LOW);
        digitalWrite(in1,HIGH);
    }
    else if(l<0)
    {
        analogWrite(enA,l);
        digitalWrite(in1,LOW);
        digitalWrite(in2,HIGH);
    }
    else
    {
        analogWrite(enA,l);
        digitalWrite(in2,LOW);
        digitalWrite(in1,LOW);
    }
}
```

```
if(r>0)
{
    analogWrite(enB,r);
    digitalWrite(in3,LOW);
    digitalWrite(in4,HIGH);
}
else if(r<0)
{
    analogWrite(enB,r);
    digitalWrite(in4,LOW);
    digitalWrite(in3,HIGH);
}
else
{
    analogWrite(enB,r);
    digitalWrite(in4,LOW);
    digitalWrite(in3,LOW);
}
}
```

Function :

ROTATE(int, int);

เราจะใส่พารามิเตอร์สองตัวเข้าไปคือค่าความเร็วของมอเตอร์ โดยค่าติดลบจะหมายถึงหมุนทวนทิศทางปกติ และอีกพารามิเตอร์คือ Delay ในการหมุน

```
void ROTATE(int s,int d)
{
    MOVE(s,-s);
    delay(d);
    MOVE(-s,s);
    delay(100);
}
```

Function :

PID();

คือฟังก์ชันหลักในการ Track เส้นทางของหุ่นยนต์ของเรา โดยฟังก์ชันจะทำการปรับค่าความเร็วของมอเตอร์ตลอดเวลา เพื่อคุมทิศทางให้ตรงเส้นดำ

```
void PID()
{
    digitalWrite(in2,LOW); digitalWrite(in3,LOW);

    if      (b2==0 and b3==1) error = 1;
    else if (b2==1 and b3==1) error = 0;
    else if (b2==1 and b3==0) error = -1;

    motorspeed = Kp*error;
    Ls = basespeed - motorspeed;
    Rs = basespeed + motorspeed;

    if(Ls > maxspeed) Ls = maxspeed;
    if(Ls < 0) Rs = basespeed;
    if(Rs > maxspeed) Ls = maxspeed;
    if(Rs < 0) Rs = basespeed;

    analogWrite(enA,Ls);
    analogWrite(enB,Rs);

    digitalWrite(in1,HIGH); digitalWrite(in4,HIGH);
}
```

Function :

CHECK();

คือฟังก์ชันที่เช็คค่าที่อ่านได้จากเซนเซอร์ ว่าเซนเซอร์ตัวไหนเจอสีอะไรอยู่

เพื่อนำไปเช็ค **case** ต่อ ว่ารถควรจะขับไปทิศทางไหน

```
void check()
{
    if(s1<200)b1=true;else b1=false;
    if(s2<200)b2=true;else b2=false;
    if(s3<200)b3=true;else b3=false;
    if(s4<200)b4=true;else b4=false;

    Serial.print(b1);Serial.print(b2);Serial.print(b3);Serial.println(b4);
}
```

STATE if else : 1-3

```
if (state == 0)
{
    if (b1 == false and b4 == true)
    {
        digitalWrite(right, 1);
        MOVE(-basespeed, -basespeed);
        delay(50);
        RESET();
        delay(50);
        state++;
    }
    MOVE(118, 100);
}
```

```
else if (state == 1) // first leaw
{
    MOVE(0, 0);
    delay(50);
    R90(-220, 300);
    while (1)
    {
        check();
        if (b2 == true and b3 == true)
        {
            MOVE(150, -150);
            delay(50);
            digitalWrite(right, 0);
            state++;
            break;
        }
        MOVE(-150, 150);
    }
}
```

```
else if (state == 2) // enter RED
{
    if (b1 == true and b2 == true and b3 == true and b4 == true)
    {
        state++;
    }
    MOVE(130, 155);
}
```

```
else if (state == 3) // Backward
{
    delay(200);
    digitalWrite(left, 1);
    digitalWrite(right, 1);
    MOVE(-basespeed, -basespeed);
    delay(50);
    RESET();
    delay(50);
    MOVE(0, 0);
    delay(50);
}
```

```
while (1)
{
    MOVE(-200, -165);
    delay(2000);
    digitalWrite(left, 0);
    state++;
    break;
}
MOVE(200, 165);
delay(50);
RESET();
delay(50);
MOVE(0, 0);
```

STATE if else : 4-5

```
else if (state == 4) //Second leaw
{
    R90(-255, 300);
    while (1)
    {
        check();
        if (b2 == true and b3 == true)
        {
            MOVE(150, -150);
            delay(50);
            digitalWrite(right, 0);
            state++;
            break;
        }
        MOVE(-155, 150);
    }
}
```

```
else if (state == 5)
{
    MOVE(100, 100);
    if (b1 == true and b4 == true)
    {
        digitalWrite(left, 1);
        MOVE(-basespeed, -basespeed);
        delay(50);
        MOVE(0, 0);
        delay(50);
        R90(150, 200);
        while (1)
        {
            check();
```

```
            if (b2 == true and b3 == true)
            {
                digitalWrite(left, 0);
                MOVE(-150, 150);
                delay(50);
                state++;
                break;
            }
            MOVE(150, -150);
        }
    }
}
```

STATE if else : 6-8

```
else if (state == 6) // go froward |_|
{
    if (b1 == true and b2 == true and b3 == true and b4 == true)
    {
        digitalWrite(left, 1);
        MOVE(-basespeed, -basespeed);
        delay(50);
        MOVE(0, 0);
        delay(50);
        state++;
    }
    MOVE(80, 80);
}
```

```
else if (state == 7) // Leaw to BLUE!!
{
    R90(150, 200);
    while (1)
    {
        check();
        if (b2 == true and b3 == true)
        {
            digitalWrite(left, 0);
            MOVE(-100, 100);
            delay(50);
            state++;
            break;
        }
        MOVE(100, -100);
    }
}
```

```
else if (state == 8)
{
    MOVE(100, 100);
    if (b1 == true and b2 == true and b3 == true and b4 == true)
    {
        digitalWrite(right, 1);
        delay(50);
        MOVE(-basespeed, -basespeed);
        delay(50);
        MOVE(0, 0);
        delay(50);
        state++;
    }
}
```


STATE if else : 9-12

```
else if (state == 9) // turn right BLUE
{
    R90(-150, 200);
    while (1)
    {
        check();
        if (b2 == true and b3 == true)
        {
            digitalWrite(right, 0);
            MOVE(150, -150);
            delay(50);
            state++;
            break;
        }
        MOVE(-150, 150);
    }
}
```

```
else if (state == 10) //backward to blue
{
    while (1)
    {
        digitalWrite(right, 1);
        digitalWrite(left, 1);
        MOVE(-200, -165);
        delay(1800);
        MOVE(200, 165);
        delay(50);
        RESET();
        delay(50);
        MOVE(0, 0);
        digitalWrite(right, 0);
        digitalWrite(left, 0);
        MOVE(80, 220);
        delay(200);
        state++;
        break;
    }
}
```

```
else if (state == 11) //before finish!!
{
    if (b1 == true and b2 == true and b3 == true and b4 == true)
    {
        state++;
    }
    MOVE(80, 80);
}
```

```
else if (state == 12)
{
    if (b1 == true and b2 == true and b3 == true and b4 == true)
    {
        MOVE(-basespeed, -basespeed);
        delay(50);
        RESET();
        delay(50);
        state++;
    }
    MOVE(80, 80);
}
```

STATE if else : 13-15

```
else if (state == 13) //forward finish
{
    MOVE(80, 80);
    if (b1 == true and b2 == true and b3 == true and b4 == true)
    {
        MOVE(-basespeed, -basespeed);
        delay(50);
        RESET();
        delay(50);
        state++;
    }
}
```

```
else if (state == 14) // turn right to FINISH!!!
{
    R90(-150, 200);
    while (1)
    {
        check();
        if (b2 == true and b3 == true)
        {
            MOVE(150, -150);
            delay(50);
            state++;
            break;
        }
        MOVE(-150, 150);
    }
}
```

```
else if (state==15)
{
    while(1)
    {

        check();
        MOVE(100, 135);
        if (b1==true and b2 == true and b3==true and b4 == true)
        {
            digitalWrite(red, 0);
            digitalWrite(green, 1);
            delay(400);
            MOVE(-basespeed, -basespeed);
            delay(50);
            RESET();
            delay(50);
            state++;
            break;
        }
    }
}
```

STATE MAP



